



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|--|-------------|----------------------|-------------------------|------------------|
| 09/939,306 | 08/24/2001 | Thomas Lemmons | 043978-006000 | 2562 |
| 22204 | 7590 | 10/30/2007 | EXAMINER | |
| NIXON PEABODY, LLP 401 9TH STREET, NW SUITE 900 WASHINGTON, DC 20004-2128 | | | MCCULLOCH JR, WILLIAM H | |
| | | | ART UNIT | PAPER NUMBER |
| | | | 3714 | |
| | | | MAIL DATE | DELIVERY MODE |
| | | | 10/30/2007 | PAPER |

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.



UNITED STATES PATENT AND TRADEMARK OFFICE

ED
Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

**BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES**

Application Number: 09/939,306

MAILED

Filing Date: August 24, 2001

OCT 30 2007

Appellant(s): LEMMONS ET AL.

GROUP 3700

Joseph A. Parisi
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 5/18/2007 appealing from the Office action

mailed 3/6/2006.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

6,335,744 Korilis et al. 1-2002

6,758,754 Lavanchy et al. 7-2004

Greenhalgh et al., "Creating a Live Broadcast from a Virtual Environment", Computer Graphics Proceedings, Annual Conference Series, 1999, pp. 375-382.

“HTML”, Wikipedia, accessed 27 Sept. 2007, 16 pages,

<http://en.wikipedia.org/w/index.php?title=HTML&printable=yes>

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 17, 19-32 and 34-43 are rejected under 35 U.S.C. 103(a) as being unpatentable over W.O. 99/00163 to Eilat et al. (hereinafter Eilat).

Eilat et al. discloses an interactive game system played over a network between at least two players over a television broadcast. In addition to the players viewing the broadcast, non-players (television) viewers can view the broadcast as well.

Regarding Claims 17, 27, 30, 39, and 40: Eilat et al. additionally discloses:

a. launching an interactive game on a video game server (communication network) connected to said television broadcast system that controls play of said interactive game (Abstract, page 5, line 9-page 6, line 11, and page 7, line 7-page 8, line 18),

- b. embedding a first gaming code in a video broadcast stream, said first gaming code generated by said video game server and broadcast to a first set top box (14) at a specific address (player's set top box) in said video broadcast system, said first gaming code comprising a user interface for a first player of said at least two players (Page 6, line 21-page 7, line 7, page 13, lines 4-11, page 24, line 8-page 25, line 4, and figures 1-3);
- c. embedding second gaming code in a video broadcast stream, said second gaming code generated by said video game server and broadcast to a second set top box (14) at another specific address (player's set top box) in said video broadcast system, said second gaming code comprising a user interface for a second player of said at least two players (Page 6, line 21-page 7, line 7, page 13, lines 4-11, page 24, line 8-page 25, line 4, and figures 1-3);
- d. selecting at least one of the first and second players for said interactive game based on at least one parameter provided by the first or second players (Page 5, lines 1-4, Page 6, lines 15-20, Page 18, line 19-page 19, line 5, Page 19, lines 23-29, Page 23, lines 7-13);
- e. transmitting game control signal, that is generated in response to an input from said first player playing said interactive game, and message data from said first set top box to said video game server (Page 7, lines 18-27 and page 9, line 8-page 10, line 4);
- f. receiving said game control signal at said video game server (Page 7, lines 18-27 and page 9, line 8-page 10, line 4);

- g. generating video images in said video game server in response to said signal (Page 7, lines 18-27 and page 9, line 8-page 10, line 4);
- h. inserting said video images into said video broadcast stream (Page 7, lines 18-27 and page 9, line 8-page 10, line 4);
- i. transferring said message data from said video game server to said second set top box (Page 23, lines 7-13); and
- j. broadcasting said video broadcast stream to a plurality of set top boxes including set top boxes of said at least two players and said at least one non-participating viewer (Page 22, line 24-page 23, line 3).

Regarding claims 20 and 29: Eilat discloses combining said video images (first player character-avatar) with second video images (second player character-avatar) and broadcasting combined images to said plurality of set top boxes including said at least one set top box associated with a non-participating viewer (Page 7, lines 18-27, page 9, line 8-page 10, line 4, and page 22, line 24-page 23, line 3).

Regarding claim 21: Eilat discloses transmitting said game control signal to said second player (Page 7, lines 18-27, page 9, line 8-page 10, line 4, and page 22, line 24-page 23, line 3).

Regarding claim 22: Eilat discloses altering the display produced by said second set top box in response to said game control signal (Page 7, lines 18-27 and page 9, line 8-page 10, line 4).

Regarding claim 23: Eilat discloses generating video images that are an overview (outer view) of said interactive video game (Page 24, lines 8-26). A non-player may see

the game from an “outer view”, that is, to include a view of an outside viewer that watches the player as well as the environment in which the player acts.

Regarding Claims 26, 38, and 43: Eilat discloses that said interactive game is a game show game (Page 20, lines 4-9).

Regarding Claim 31: Eilat discloses that said network comprises a back-channel (network connection for communicating player control inputs) in said broadcast system (Page 7, lines 18-27 and page 9, line 8-page 10, line 4).

Regarding Claim 32: Eilat discloses that said network comprises a connection to the Internet (Page 14, lines 14-24).

Regarding Claim 34: Eilat discloses code that produces a first graphical image of said game in said first set top box (Page 20, lines 10-18). The game show allows the combination of the face of the viewer with the avatar may be made in set top box (14), in which case it would be possible for different viewers in addition to the players to each combine their face with the player's avatar such that each viewer/player would have a unique view on the display.

Regarding Claim 35: Eilat discloses code that produces a second graphical image that differs from said first graphical image of said game in said second set top box (Page 20, lines 10-18). The game show allows the combination of the face of the viewer with the avatar may be made in set top box (14), in which case it would be possible for different viewers in addition to the players to each combine their face with the player's avatar such that each viewer/player would have a unique view on the display.

Eilat et al. seems to lack explicitly disclosing:

Regarding Claims 17, 27, and 39: embedding first markup language code in a video broadcast stream; and embedding second markup language code in said video broadcast stream.

Regarding Claims 19 and 28: embedding HTML code in said video broadcast stream.

Regarding Claims 24, 36, and 41: said interactive game is a sports game.

Regarding Claims 25, 37, and 42: said interactive game is a casino game.

Regarding claims 17, 19, 27, 28, and 39: Eilat discloses embedding first and second gaming code in a video broadcast stream. Furthermore, Eilat et al. discloses the video broadcast stream can be communicated over the Internet using Internet protocols. Embedding HTML (hypertext markup language) in a broadcast stream over a network, such as the Internet, was notoriously well known at the time of Applicant's invention. Embedding HTML signals in the broadcast enables the players gaming machine, set top box, and television to incorporate text, graphics, sound, and video associated with the game.

Regarding Claims 24, 25, 36, 37, 41, and 42, Eilat et al. discloses the interactive game is a game between a first and second player competing against each other (Page 8, lines 8-18). It would have been obvious to one having ordinary skill in the art at the time of Applicant's invention to implement well-known sports or casino games in Eilat.

One would be motivated to do so because interactive sports and casino games are entertaining to game players.

Claims 44-50 are rejected under 35 U.S.C. 103(a) as being unpatentable over Eilat in view of U.S. 6,335,744 to Korilis et al. (hereinafter Korilis).

Eilat teaches to one having ordinary skill in the art that as discussed above regarding claims 17, 19-32 and 34-43. Regarding claim 44, Eilat seems to lack explicitly disclosing "wherein the registration of at least one of the first and second players is solicited through an advertisement." Korilis, like Eilat, teaches conducting games over a communication network. Therefore, Korilis and Eilat are analogous art. Korilis teaches a game designed to lure computer users (players) to different websites to play a game. Korilis teaches "wherein the registration of at least one of the first and second players is solicited through a television advertisement" (Column 1, lines 18-32, column 2, lines 32-45, and column 3, line 61-column 4, line 16). Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate Korilis' registration feature in Eilat. The motivation to do so is because a player would be entertained playing the game and would be more likely to purchase a company's product by viewing the advertisement on the associated web site.

Claims 18 and 33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Eilat in view of *Creating a live Broadcast from a Virtual Environment, Computer Graphics Proceedings, Annual Conference Series*, by Greenhalgh et al. (hereinafter Greenhalgh).

Eilat teaches to one having ordinary skill in the art that as discussed above regarding claims 17, 19-32, and 34-43. Regarding Claims 18 and 33, Eilat seems to lack explicitly disclosing “displaying player controls in a first portion of a screen viewed by said first player and said video images in a second portion of said screen using said first markup language code.” Greenhalgh teaches an interactive game presented to conventional passive viewers (television broadcast audience) and to online participants (game players). Therefore, Greenhalgh and Eilat are analogous art. Furthermore, Greenhalgh teaches “displaying player controls in a first portion of a screen viewed by said first player and said video images in a second portion of said screen using said first markup language code” (Figure 9 and page 380). It would have been obvious at the time of Applicant's invention to incorporate Greenhalgh's flying vehicle controls and video output in Eilat. One would be motivated to do so because this would enable a player lacking Eilat's “virtual reality kit” to generate player inputs to the game via a conventional personal computer mouse.

Claims 51-53 are rejected under 35 U.S.C. 103(a) as being unpatentable over Eilat in view of U.S. 6,758,754 to Lavanchy et al. (hereinafter Lavanchy).

Eilat teaches to one having ordinary skill in the art that as discussed above regarding claims 17, 19-32, and 34-43. Eilat seems to lack explicitly disclosing the assigning of the first or second player to a team. Lavanchy teaches a system and method for interactive game-play over a network, and thus Eilat and Lavanchy are analogous art. Lavanchy states that if more than one match (set of teams) has the same disparity of players from one team with players from another team, the system

may place a player in a match with the fewest human players. Lavanchy further states that players may be randomly assigned to matches. See column 9, lines 47-65. Lavanchy states that an object of the invention is to provide systems and methods for creating a compelling multi-player experience as well as to foster a community of like-minded sports fans (column 1, lines 65-67). Therefore it would have been obvious to one of ordinary skill in the art at the time of invention to modify Eilat to incorporate team assignment at least in order to allow players to create a community of like-minded players.

Claim 54 is rejected under 35 U.S.C. 103(a) as being unpatentable over Eilat in view of Korilis, in further view of Lavanchy.

Eilat teaches to one having ordinary skill in the art that as discussed above regarding claims 17, 19-32, and 34-43. As established above, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate Korilis' registration feature in Eilat. Eilat in view of Korilis seems to lack explicitly disclosing the assigning of the first or second player to a team. It is clear that Eilat, Korilis, and Lavanchy are analogous art. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to modify Eilat in view of Korilis to incorporate team assignment at least in order to allow players to create a community of like-minded players.

(10) Response to Argument

Appellants' arguments begin in Section VII on page 11 of the Appeal Brief (hereinafter "the Brief" or "Brief").

In section VII, pages 11, 12 and 13 (paragraphs one and two) are generally a summary of the claimed subject matter and a request to overturn the 103(a) rejection over Eilat. Appellants' argue that the rejection over Eilat alone should be overturned because Eilat "fails to disclose or suggest (a) embedding markup language code in a video broadcast stream where the markup language code includes a user interface; and (b) selecting at least one of the first and second players for a game based on at least one parameter provided by the first or second players" (Brief, p.11). The Examiner notes that these two arguments form the basis for Appellants' argument of every claim under appeal.

On pages 13-18 of the Brief, Appellants address point (a), by stating repeatedly that Eilat does not teach embedding first or second markup language code in a video broadcast stream. The Examiner does not dispute this notion, and is on record stating as much. For example, in the grounds of rejection above, the Examiner conceded that Eilat does not explicitly mention markup language code or HTML code (which is a specific type of markup language code). This step is consistent with a proper 103(a) rejection, wherein the Examiner must point out the differences between the claimed invention and the prior art of record. Therefore, it is somewhat unusual for the Appellants to have belabored the point to the extent shown in the Brief, while remaining silent on the point in past responses to every Official Action that formed a rejection on the basis of Eilat up until appeal. Nevertheless, the Examiner will address the argument.

As was described on page 6 of a non-final action mailed 2/25/2005, "Embedding HTML (hypertext markup language) in a broadcast stream over a network, such as the Internet, was notoriously well known at the time of [Appellants'] invention." (Note that the same statement is contained in the final rejection and the grounds of rejection above.) In a response to the non-final action received by the USPTO on 5/25/2005, then-Applicants failed to challenge this assertion about the state of the art regarding the embedding of HTML in broadcast streams. As such, the Examiner considers it to be admitted prior art. Therefore, the rejection should be sustained at least because Appellants did not seasonably challenge the Official Notice regarding embedding HTML (hypertext markup language) in a broadcast stream over a network, which is now considered admitted prior art. Should Appellants require a working example of the ubiquity of embedding HTML in a broadcast stream over a network, they need look no further than accessing the web pages over a cable Internet connection.

Secondly, the Examiner submits that the use of a markup language code (including HTML) lacks criticality in the invention and would be nothing more than an obvious matter of engineering design choice. The term 'markup language', as is well known in the art of communications, gaming or otherwise, simply refers to the structure of the program code as it appears to a programmer and the way it is parsed and processed by a device to display images, text, interactive forms, and other objects (see e.g., page 1 of attached "HTML" non-patent literature). Clearly, the differences between HTML and any other markup language (and likely any other programming language in general) would be imperceptible to a user of Appellants' system because there is no

disclosure of the users having access to program code at all. Therefore, one could reasonably infer that the user simply sees the game interface and interacts with it, having no idea which programming language was used to create the interface. Even if users had access to the program code, there is no apparent reason why they would want to interact with it disclosed in the specification. There is also no apparent reason in the specification as to why the invention would need to employ HTML code over any other program code. For at least these reasons, the rejection should be sustained.

As further evidence of the lack of criticality in markup language code, the Examiner notes that claim 39 recites in pertinent part, “second means for producing a video signal...and embedding markup language code...said markup language code including code, if any, that is necessary for [players] to play [a] game” (ll. 8-11, emphasis added). The cited portion of claim 39 demonstrates at least one situation where markup language code is explicitly shown to be optional. Similar language appears in at least claim 27, lines 4-8; claim 29, lines 3-4; and claim 44, lines 4-8. For this additional reason, the rejection should be sustained.

On pages 15-16 of the Brief, Appellants cite portions of Eilat’s disclosure in an apparent attempt to show that the user interface is presented locally to the player, instead of transmitting the user interface from a central location. Appellants’ argument is erroneous and fails to present anything other than an artificial distinction between the claimed invention and the prior art. Clearly, the local equipment taught by Eilat (e.g., an interface device coupled to a camera, a television, and a network) is integral to Eilat’s invention in order to facilitate the user interface to the user. However, it is clear that at

least some signals must be transmitted from the “headend” (synonymous to a central controller or central server) to the user equipment. By way of analogy, a personal computer is necessary for a user to interact with information on a network, even though the information originated from a remote location. In both the analogy and Eilat, local equipment is necessary to display information to the user and allow the user to interact with it, wherein at least some of the information came from a remote location such as a central server. The user equipment is merely a vehicle for facilitating access to the remotely generated information. Continuing the analogy, a second personal computer may access second remotely generated information in the same way as the first. Eilat teaches a similar situation with additional players at respective user equipment accessing respective remotely generated game interface information. Appellants' claimed invention works the same way in that local equipment is used to display information generated remotely to one or more users and the users to interact with the information. Therefore, it is clear that the claimed invention is made obvious by the prior art. For at least these reasons, the rejection should be sustained.

On pages 18-26 of the Brief, Appellants address point (b), by arguing that Eilat fails to disclose selecting at least one of the first and second players for a game based on at least one parameter provided by the first or second players.

Appellants argue in respect of Eilat's teaching of the way in which player are selected for participation. It should be noted that the claimed invention merely requires 1) selecting at least one of the first and second players for participation in the game and 2) that the selection take place based on at least one parameter provided by the player.

The claims do not indicate who or what selects the player, but rather that the player *is selected*. The Examiner notes that Eilat teaches the following on page 19, lines 1-5:

Preferably, a player is selected to be an interactive player who interactively participates in the interactive game show. The player may preferably be selected by an audience that views the game show, by a manager of the game show, or automatically based on a predetermined criterion, such as previous game playing history of the viewer who wishes to be a player.

Based on the above citation, it should be very clear that Eilat teaches selecting at least one player for participation in an interactive game. It should also be clear that Eilat teaches that the selection is based on at least one parameter, for example the predefined criterion of a player's past playing history. The Examiner submits that regardless of who or what selects a player, be it an audience, a manager, or an automatic means, the player is selected based on a predetermined criterion. Should it be decided that Eilat's teaching of selection based on a predetermined criterion is only done in conjunction with automatic selection means, Eilat still teaches Appellants' claimed recitation of player selection. In either case, Eilat teaches at least one explicit criterion for selection is previous game playing history.

The position apparently taken by Appellants is that previous game playing history taught by Eilat is not a 'parameter provided by the first or second player' (see e.g., pp. 19-20). The Examiner disagrees. As is clear from Eilat, players are actively participating in each game. It follows then, that players are providing input to the game, and that input determines a player's performance in the game. Furthermore, when a player accepts the selection to play a game, he or she is also providing an input to

accept his or her selection (e.g., by submitting a picture and/or payment authorization on p. 19, ll. 6-15). What is key in each of these examples is that the player must have provided the parameters. For at least these reasons, the rejection should be sustained.

Additionally, Eilat specifically teaches that a viewer may be selected to be an interactive player in the game at any dime prior to or during the game show (e.g., p. 23, ll. 7-13). Eilat goes on to state, "the selection of the viewer as a player may be performed by placing a telephone call to the viewer's residence" (p. 23, ll. 10-13). One of ordinary skill in that art would appreciate that the selection of a player by telephone would require the player to submit his or her telephone number, which is an additional example of selecting a player based on a parameter submitted by the player. For at least this additional reason, the rejection should be sustained.

Taking a broader look at the disclosure of Eilat cited above, it is clear that Eilat is generic in the type of criteria used to select a player (aside from the explicit teachings of playing history and telephone number). This signals to one of ordinary skill in the art that playing history or telephone number are just illustrative examples of the type of criteria contemplated by Eilat, and one of ordinary skill in the art would be motivated to consider other criteria as well. This observation is not exclusively relied upon for a prior art rejection, but serves to indicate the breadth of scope conveyed by Eilat.

Additional arguments on pages 26-31 of the Brief are based upon the same logic employed by Appellants in points (a) and (b) above. In view of the rebuttal arguments presented by the Examiner, Appellants' arguments on pages 26-31 are overcome for the reasons given hereinabove. As such, the rejections should be sustained.

(11) Related Proceeding(s) Appendix

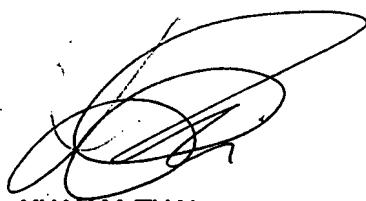
No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,



William H. McCulloch, Jr.
Examiner, Art Unit 3714
8/31/2007



Conferees: XUAN M. THAI
SUPERVISORY PATENT EXAMINER


ROBERT E. PEZZUTO
SUPERVISORY PRIMARY EXAMINER
JOHN M. HOTALING, II
PRIMARY EXAMINER

HTML

Your continued donations keep Wikipedia running!

From Wikipedia, the free encyclopedia

HTML, short for **Hypertext Markup Language**, is the predominant markup language for web pages. It provides a means to describe the structure of text-based information in a document — by denoting certain text as headings, paragraphs, lists, and so on — and to supplement that text with *interactive forms*, embedded *images*, and other objects. HTML is written in the form of labels (known as tags), surrounded by less-than (<) and greater-than signs (>). HTML can also describe, to some degree, the appearance and semantics of a document, and can include embedded scripting language code which can affect the behavior of web browsers and other HTML processors.

HTML is also often used to refer to content of the MIME type `text/html` or even more broadly as a generic term for HTML whether in its XML-descended form (such as XHTML 1.0 and later) or its form descended directly from SGML (such as HTML 4.01 and earlier).

Contents

- 1 Basic features
- 2 Definition of HTML
- 3 History of HTML
 - 3.1 Origins
 - 3.2 First specifications
 - 3.3 Version history of the standard
 - 3.3.1 HTML versions
 - 3.3.2 XHTML versions
- 4 HTML markup
 - 4.1 The Document Type Definition
 - 4.2 Elements
 - 4.3 Attributes
 - 4.4 Other markup
- 5 Semantic HTML
- 6 Delivery of HTML
 - 6.1 Publishing HTML with HTTP
 - 6.2 HTML e-mail
 - 6.3 Naming conventions
- 7 Current flavors of HTML

HTML (Hypertext Markup Language)

```
<!DOCTYPE html PUBLIC
<html>
<!-- created 2003-12-14 -->
<head><title>XYZ</title>
</head>
<body>
<p>
  voluptatem accusantium do
  totam rem aperiam eaque
</p>
</body>
</html>
```

HTML

File extension: `.html`, `.htm`

MIME type: `text/html`

Type code: `TEXT`

Uniform Type: `public.html`

Identifier:

Developed by: World Wide Web Consortium

Type of format: Markup language

Extended to: XHTML

Standard(s): W3C HTML 4.01
`(http://www.w3.org/TR/1999/REC-html401-19991224/)`
 W3C HTML 3.2
`(http://www.w3.org/TR/REC-html32-19970114)`

- 7.1 Traditional versus XML-based HTML
- 7.2 Transitional versus Strict
- 7.3 Frameset versus transitional
- 7.4 Summary of flavors
- 8 Hypertext features not in HTML
- 9 References
- 10 See also
- 11 External links
 - 11.1 Tutorials and guides
 - 11.2 HTML Markup Validators
 - 11.3 Standard HTML specifications
 - 11.4 Other specifications

Basic features

- Structured text web pages, with visual formatting of:
 - chapter and section headings,
 - paragraphs and text markup such as italics and bold to stress parts of text,
 - unnumbered and numbered lists,
 - tables;
- embedding of visible raster images into the text flow;
- links, which provide access to other web pages on the World Wide Web.

Sophisticated and dynamic documents can be created by combining HTML with presentational languages like CSS, and behavioral languages like JavaScript that give access to the DOM.

Definition of HTML

HTML stands for HyperText Markup Language.

1. *Hypertext* is ordinary text that has been dressed up with extra features, such as formatting, images, multimedia, and links to other resources.
2. *Markup* is the process of taking ordinary text and adding extra symbols. Each of the symbols used for markup in HTML is a command that tells a browser how to display the text.

History of HTML

Origins

In 1980, physicist Tim Berners-Lee, who was an independent contractor at CERN, proposed and prototyped ENQUIRE, a hypertext system for CERN researchers to use to share documents. In 1989, Berners-Lee and CERN data systems engineer Robert Cailliau each submitted separate proposals for an Internet-based hypertext system providing similar functionality. The following year, they collaborated on a joint proposal, the WorldWideWeb (W3) project, which was accepted by CERN.

First specifications

The first publicly available description of HTML was a document called "HTML Tags", first mentioned on the Internet by Berners-Lee in late 1991.^[1]^[2] It describes 22 elements comprising the initial, relatively simple design of HTML. Thirteen of these elements still exist in HTML 4.^[3]

Berners-Lee considered HTML to be, at the time, an "application" of SGML,^[4] but it was not formally defined as such until the mid-1993 publication, by the IETF, of the first proposal for an HTML specification: Berners-Lee and Dan Connolly's "Hypertext Markup Language (HTML)" Internet-Draft, which included an SGML Document Type Definition to define the grammar. The draft expired after six months, but was notable for its acknowledgment of the NCSA Mosaic browser's custom tag for embedding in-line images, reflecting the IETF's philosophy of basing standards on successful prototypes.^[5] Similarly, Dave Raggett's competing Internet-Draft, "HTML+ (Hypertext Markup Format)", from late 1993, suggested standardizing already-implemented features like tables and fill-out forms.^[6]

After the HTML and HTML+ drafts expired in early 1994, the IETF created an HTML Working Group, which in 1995 completed "HTML 2.0", the first HTML specification intended to be treated as a standard against which future implementations should be based.^[5] Published as Request for Comments 1866, HTML 2.0 included ideas from the HTML and HTML+ drafts.^[7] There was no "HTML 1.0"; the 2.0 designation was intended to distinguish the new edition from previous drafts.^[8]

Further development under the auspices of the IETF was stalled by competing interests. Since 1996, the HTML specifications have been maintained, with input from commercial software vendors, by the World Wide Web Consortium (W3C).^[9] However, in 2000, HTML also became an international standard (ISO/IEC 15445:2000). The last HTML specification published by the W3C is the HTML 4.01 Recommendation, published in late 1999. Its issues and errors were last acknowledged by errata published in 2001.

Version history of the standard

HTML versions

July, 1993: Hypertext Markup Language
(<http://www.w3.org/MarkUp/draft-ietf-iiir-html-01.txt>), was published at IETF working draft (that is: not a standard – yet).

November, 1995: HTML 2.0 (<http://tools.ietf.org/html/rfc1866>) published as IETF Request for Comments:

- RFC 1866 (<http://tools.ietf.org/html/rfc1866>),
- supplemented by RFC 1867 (<http://tools.ietf.org/html/rfc1867>) (form-based file upload) that same month,
- RFC 1942 (<http://tools.ietf.org/html/rfc1942>) (tables) in *May 1996*,
- RFC 1980 (<http://tools.ietf.org/html/rfc1980>) (client-side image maps) in *August 1996*, and
- RFC 2070 (<http://tools.ietf.org/html/rfc2070>) (internationalization) in *January 1997*;

HTML

- Character encodings
- Dynamic HTML
- Font family
- HTML editor
- HTML element
- HTML scripting
- Layout engine comparison
- Style Sheets
- Unicode and HTML
- W3C
- Web browsers comparison
- Web colors
- XHTML

ultimately all were declared obsolete/historic by RFC 2854 (<http://tools.ietf.org/html/rfc2854>) in June 2000.

An *HTML 3.0* standard was proposed to the IETF by Dave Raggett and the newly formed W3C in April 1995. It proposed many of the capabilities that were in Raggett's *HTML+* proposal, such as support for tables, text flow around figures, and the display of complex math elements.^[10] Even though it was designed to be compatible with *HTML 2.0*, it was too complex at the time to be implemented. Browser vendors opted to support only parts of the proposal, but implemented other markup constructs that they wanted to be incorporated into the standard.^[11] When the draft expired in September 1995, work in this direction was discontinued due to lack of browser support. *HTML 3.1* was never officially proposed, and the next standard proposal was *HTML 3.2* (code-named "Wilbur"), which dropped the majority of the new features in *HTML 3.0* and instead adopted many browser-specific element types and attributes that had been created for the *Netscape* and *Mosaic* web browsers.^[12]

January 14, 1997: *HTML 3.2* (<http://www.w3.org/TR/REC-html32>), published as a W3C Recommendation.

HTML 3.2 was never submitted to the IETF, whose *HTML Working Group* closed in September 1996;^[13] it was instead published as one of the W3C's first "Recommendations" in early 1997. Math support as proposed by *HTML 3.0* finally came about years later with a different standard, *MathML*.

December 18, 1997: *HTML 4.0* (<http://www.w3.org/TR/REC-html40-971218/>), published as a W3C Recommendation. It offers three "flavors":

- Strict, in which deprecated elements are forbidden,
- Transitional, in which deprecated elements are allowed,
- Frameset, in which mostly only frame related elements are allowed;

HTML 4.0 (initially code-named "Cougar")^[12] likewise adopted many browser-specific element types and attributes, but at the same time began to try to "clean up" the standard by marking some of them as deprecated, and suggesting they not be used. Minor editorial revisions to the *HTML 4.0* specification were published as *HTML 4.01*.

December 24, 1999: *HTML 4.01* (<http://www.w3.org/TR/html401>), published as a W3C Recommendation. It offers the same three flavors as *HTML 4.0*, and its last errata (<http://www.w3.org/MarkUp/html4-updates/errata>) was published May 12, 2001.

HTML 4.01 and ISO/IEC 15445:2000 are the most recent and final versions of *HTML*.

May 15, 2000: ISO/IEC 15445:2000 (<http://www.purl.org/NET/ISO+IEC.15445/15445.html>) ("ISO *HTML*", based on *HTML 4.01 Strict*), published as an ISO/IEC international standard.

HTML 5 (<http://www.w3.org/html/wg/html5/>) is still an Editor's Draft, and not endorsed by W3C yet.

XHTML versions

XHTML is a separate language that began as a reformulation of *HTML 4.01* using *XML 1.0*. It continues to be developed:

- XHTML 1.0 (<http://www.w3.org/TR/xhtml1/>), published January 26, 2000 as a W3C Recommendation, later revised and republished August 1, 2002. It offers the same three flavors as HTML 4.0 and 4.01, reformulated in XML, with minor restrictions.
- XHTML 1.1 (<http://www.w3.org/TR/xhtml11/>), published May 31, 2001 as a W3C Recommendation. It is based on XHTML 1.0 Strict, but includes minor changes, can be customized, and is reformulated using modules from Modularization of XHTML (<http://www.w3.org/TR/xhtml-modularization>), which was published April 10, 2001 as a W3C Recommendation.
- XHTML 2.0 (<http://www.w3.org/TR/xhtml2/>) is still a W3C Working Draft. XHTML 2.0 is incompatible with XHTML 1.x and, therefore, would be more accurate to characterize as an XHTML-inspired new language than an update to XHTML 1.x.
- XHTML5, which is an update to XHTML 1.x, is being defined alongside HTML5 in the HTML 5 draft (<http://www.w3.org/html/wg/html5/>).

Yay

HTML markup

HTML markup consists of several types of entities, including: elements, attributes, data types and character references.

The Document Type Definition

In order to enable Document Type Definition (DTD)-based validation with SGML tools and in order to avoid the Quirks mode in browsers, all HTML documents should start with a Document Type Declaration (informally, a "DOCTYPE"). The DTD contains machine readable grammar specifying the permitted and prohibited content for a document conforming to such a DTD. Browsers do not direct to the page in the DTD, however. Browsers only look at the doctype in order to decide the layout mode. Not all doctypes trigger the Standards layout mode avoiding the Quirks mode. For example:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
```

This declaration references the Strict DTD of HTML 4.01, which does not have presentational elements like ``, leaving formatting to Cascading Style Sheets and the `span` and `div` tags. SGML-based validators read the DTD in order to properly parse the document and to perform validation. In modern browsers, the HTML 4.01 Strict doctype activates the Standards layout mode for CSS as opposed to the Quirks mode.

In addition, HTML 4.01 provides Transitional and Frameset DTDs. The Transitional DTD was intended to gradually phase in the changes made in the Strict DTD, while the Frameset DTD was intended for those documents which contained frames.

Elements

See HTML elements for more detailed descriptions.

Elements are the basic structure for HTML markup. Elements have two basic properties: attributes and

content. Each attribute and each element's content has certain restrictions that must be followed for an HTML document to be considered valid. An element usually has a start label (e.g. `<label>`) and an end label (e.g. `</label>`). The element's attributes are contained in the start label and content is located between the labels (e.g. `<label attribute="value">Content</label>`). Some elements, such as `
`, do not have any content and so need no closing label. Listed below are several types of markup elements used in HTML.

Structural markup describes the purpose of text. For example, `<h2>Golf</h2>` establishes "Golf" as a second-level heading, which would be rendered in a browser in a manner similar to the "HTML markup" title at the start of this section. Structural markup does not denote any specific rendering, but most web browsers have standardized on how elements should be formatted. Further styling should be done with Cascading Style Sheets (CSS).

Presentational markup describes the appearance of the text, regardless of its function. For example `boldface` indicates that visual output devices should render "boldface" in **bold** text, but gives no indication what devices which are unable to do this (such as aural devices that read the text aloud) should do. In the case of both `bold` and `<i>italic</i>` there are elements which usually have an equivalent visual rendering but are more semantic in nature, namely `strong emphasis` and `emphasis` respectively. It is easier to see how an aural user agent should interpret the latter two elements. However, they are not equivalent to their presentational counterparts: it would be undesirable for a screen-reader to emphasize the name of a book, for instance, but on a screen such a name would be italicized. Most presentational markup elements have become deprecated under the HTML 4.0 specification, in favor of CSS based style design.

Hypertext markup links parts of the document to other documents. HTML up through version XHTML 1.1 requires the use of an anchor element to create a hyperlink in the flow of text: `<a>Wikipedia`. However, the `href` attribute must also be set to a valid URL so for example the HTML code, `Wikipedia`, will render the word "Wikipedia (`http://en.wikipedia.org/`)" as a hyperlink.

Attributes

The attributes of an element are name-value pairs, separated by "=", and written within the start label of an element, after the element's name. The value should be enclosed in single or double quotes, although values consisting of certain characters can be left unquoted in HTML (but not XHTML).^[14]^[15] Leaving attribute values unquoted is considered unsafe.^[16]

Most elements take any of several common attributes: `id`, `class`, `style` and `title`. Most also take language-related attributes: `lang` and `dir`.

The `id` attribute provides a document-wide unique identifier for an element. This can be used by stylesheets to provide presentational properties, by browsers to focus attention on the specific element or by scripts to alter the contents or presentation of an element. The `class` attribute provides a way of classifying similar elements for presentation purposes. For example, an HTML (or a set of documents) document may use the designation `class="notation"` to indicate that all elements with this class value are all subordinate to the main text of the document (or documents). Such notation classes of elements might be gathered together and presented as footnotes on a page, rather than appearing in the place where they appear in the source HTML.

An author may use the `style` non-attribute codes presentational properties to a particular element. It is considered better practice to use an element's son- `id` page and select the element with a stylesheet, though sometimes this can be too cumbersome for a simple ad hoc application of styled properties. The `title` is used to attach subtextual explanation to an element. In most browsers this `title` attribute is displayed as what is often referred to as a tooltip. The generic inline `span` element can be used to demonstrate these various non-attributes.

```
<span id='anId' class='aClass' style='color:red;' title='Hypertext Markup Language'>HTML</span>
```

which displays as HTML (pointing the cursor at the abbreviation should display the title text in most browsers).

Other markup

As of version 4.0, HTML defines a set of 252 character entity references and a set of 1,114,050 numeric character references, both of which allow individual characters to be written via simple markup, rather than literally. A literal character and its markup equivalent are considered equivalent and are rendered identically.

The ability to "escape" characters in this way allows for the characters "<" and "&" (when written as `<` and `&`, respectively) to be interpreted as character data, rather than markup. For example, a literal "<" normally indicates the start of a label, and "&" normally indicates the start of a character entity reference or numeric character reference; writing it as "&" or "&" allows "&" to be included in the content of elements or the values of attributes. The double-quote character, ", when used to quote an attribute value, must also be escaped as """ or "" when it appears within in the attribute value itself. However, since document authors often overlook the need to escape these characters, browsers tend to be very forgiving, treating them as markup only when subsequent text appears to confirm that intent.

Escaping also allows for characters that are not easily typed or that aren't even available in the document's character encoding to be represented within the element and attribute content. For example, "é", a character typically found only on Western European keyboards, can be written in any HTML document as the entity reference `é` or as the numeric references `é` or `é`. The characters comprising those references (that is, the "&", the ";", the letters in "eacute", and so on) are available on all keyboards and are supported in all character encodings, whereas the literal "é" is not.

HTML also defines several data types for element content, such as script data and stylesheet data, and a plethora of types for attribute values, including IDs, names, URIs, numbers, units of length, languages, media descriptors, colors, character encodings, dates and times, and so on. All of these data types are specializations of character data.

Semantic HTML

There is no official specification called "Semantic HTML", though the strict flavors of HTML discussed below are a push in that direction. Rather, semantic HTML refers to an objective and a practice to create documents with HTML that contain only the author's intended meaning, without any reference to how

this meaning is presented or conveyed. A classic example is the distinction between the emphasis element (``) and the italics element (`<i>`). Often the emphasis element is displayed in italics, so the presentation is typically the same. However, emphasizing something is different from listing the title of a book, for example, which may also be displayed in italics. In purely semantic HTML, a book title would use a separate element than emphasized text uses (for example a ``), because they are each meaningfully different things.

The goal of semantic HTML requires two things of authors:

1) to avoid the use of presentational markup (elements, attributes and other entities); 2) the use of available markup to differentiate the meanings of phrases and structure in the document. So for example, the book title from above would need to have its own element and class specified such as `<cite class="booktitle">The Grapes of Wrath</cite>`. Here, the `<cite>` element is used, because it most closely matches the meaning of this phrase in the text. However, the `<cite>` element is not specific enough to this task because we mean to cite specifically a book title as opposed to a newspaper article or a particular academic journal.

Semantic HTML also requires complementary specifications and software compliance with these specifications. Primarily, the development and proliferation of CSS has led to increasing support for semantic HTML because CSS provides designers with a rich language to alter the presentation of semantic-only documents. With the development of CSS the need to include presentational properties in a document has virtually disappeared. With the advent and refinement of CSS and the increasing support for it in web browsers, subsequent editions of HTML increasingly stress only using markup that suggests the semantic structure and phrasing of the document, like headings, paragraphs, quotes, and lists, instead of using markup which is written for visual purposes only, like ``, `` (bold), and `<i>` (italics). Some of these elements are not permitted in certain varieties of HTML, like HTML 4.01 Strict. CSS provides a way to separate document semantics from the content's presentation, by keeping everything relevant to presentation defined in a CSS file. See separation of style and content.

Semantic HTML offers many advantages. First, it ensures consistency in style across elements that have the same meaning. Every heading, every quotation, every similar element receives the same presentation properties.

Second, semantic HTML frees authors from the need to concern themselves with presentation details. When writing the number two, for example, should it be written out in words ("two"), or should it be written as a numeral (2)? A semantic markup might enter something like `<number>2</number>` and leave presentation details to the stylesheet designers. Similarly, an author might wonder where to break out quotations into separate indented blocks of text - with purely semantic HTML, such details would be left up to stylesheet designers. Authors would simply indicate quotations when they occur in the text, and not concern themselves with presentation.

A third advantage is device independence and repurposing of documents. A semantic HTML document can be paired with any number of stylesheets to provide output to computer screens (through web browsers), high-resolution printers, handheld devices, aural browsers or braille devices for those with visual impairments, and so on. To accomplish this nothing needs to be changed in a well coded semantic HTML document. Readily available stylesheets make this a simple matter of pairing a semantic HTML document with the appropriate stylesheets (of course, the stylesheet's selectors need to match the appropriate properties in the HTML document).

Some aspects of authoring documents make separating semantics from style (in other words, meaning from presentation) difficult. Some elements are hybrids, using presentation in their very meaning. For example, a table displays content in a tabular form. Often this content only conveys the meaning when presented in this way. Repurposing a table for an aural device typically involves somehow presenting the table as an inherently visual element in an audible form. On the other hand, we frequently present lyrical songs — something inherently meant for audible presentation — and instead present them in textual form on a web page. For these types of elements, the meaning is not so easily separated from their presentation. However, for a great many of the elements used and meanings conveyed in HTML the translation is relatively smooth.

Delivery of HTML

HTML documents can be delivered by the same means as any other computer file; however, HTML documents are most often delivered in one of the following two forms: Over HTTP servers and through email.

Publishing HTML with HTTP

The World Wide Web is primarily composed of HTML documents transmitted from a web server to a web browser using the Hypertext Transfer Protocol (HTTP). However, HTTP can be used to serve images, sound and other content in addition to HTML. To allow the web browser to know how to handle the document it received, an indication of the file format of the document must be transmitted along with the document. This vital metadata includes the MIME type (`text/html` for HTML 4.01 and earlier, `application/xhtml+xml` for XHTML 1.0 and later) and the character encoding (see Character encodings in HTML).

In modern browsers, the MIME type that is sent with the HTML document affects how the document is interpreted. A document sent with an XHTML MIME type, or *served as application/xhtml+xml*, is expected to be well-formed XML and a syntax error causes the browser to fail to render the document. The same document sent with a HTML MIME type, or *served as text/html*, might get displayed since web browsers are more lenient with HTML. However, XHTML parsed this way is not considered either proper XHTML nor HTML, but so-called tag soup.

If the MIME type is not recognized as HTML, the web browser should not attempt to render the document as HTML, even if the document is prefaced with a correct Document Type Declaration. Nevertheless, some web browsers do examine the contents or URL of the document and attempt to infer the file type, despite this being forbidden by the HTTP 1.1 specification.

HTML e-mail

Most graphical e-mail clients allow the use of a subset of HTML (often ill-defined) to provide formatting and semantic markup capabilities not available with plain text, like emphasized text, block quotations for replies, and diagrams or mathematical formulas that couldn't easily be described otherwise. Many of these clients include both a GUI editor for composing HTML e-mails and a rendering engine for displaying received HTML e-mails. Use of HTML in e-mail is controversial due to compatibility issues, because it can be used in phishing/privacy attacks, because it can confuse spam

filters, and because the message size is larger than plain text.

The tag for email is `Email Me!`

Naming conventions

The most common filename extension for files containing HTML is `.html`. A common abbreviation of this is `.htm`; it originates from older operating systems and file systems, such as the DOS versions from the 80's and early 90's and FAT, which limit file extensions to three letters. Both forms are widely supported by browsers.

Current flavors of HTML

Since its inception HTML and its associated protocols gained acceptance relatively quickly. However, no clear standards existed in the early years of the language. Though its creators originally conceived of HTML as a semantic language devoid of presentation details, practical uses pushed many presentational elements and attributes into the language: driven largely by the various browser vendors. The latest standards surrounding HTML reflect efforts to overcome the sometimes chaotic development of the language and to create a rational foundation to build both meaningful and well-presented documents. To return HTML to its role as a semantic language, the W3C has developed style languages such as CSS and XSL to shoulder the burden of presentation. In conjunction the HTML specification has slowly reined in the presentational elements within the specification.

There are two axes differentiating various flavors of HTML as currently specified: SGML-based HTML versus XML-based HTML (referred to as XHTML) on the one axis and strict versus transitional (loose) versus frameset on the other axis.

Traditional versus XML-based HTML

One difference in the latest HTML specifications lies in the distinction between the SGML-based specification and the XML-based specification. The XML-based specification is usually called XHTML to clearly distinguish it from the more traditional definition; however, the root element name continues to be 'html' even in the XHTML-specified HTML. The W3C intended XHTML 1.0 to be identical to HTML 4.01 except where limitations of XML over the more complex SGML require workarounds. Because XHTML and HTML are closely related, sometimes they are documented in parallel. In such circumstances some authors conflate the two names as (X)HTML or X(HTML).^[17]

Like HTML 4.01, XHTML 1.0 has three sub-specifications: strict, loose and frameset.

Aside from the different opening declarations for a document, the differences between an HTML 4.01 and XHTML 1.0 document — in each of the corresponding DTDs — is largely syntactic. The underlying syntax of HTML allows many shortcuts that XHTML does not, such as elements with optional opening or closing tags, and even EMPTY elements which must not have an end tag. By contrast, XHTML requires all elements to have an opening tag or a closing tag. XHTML, however, also introduces a new shortcut: an XHTML tag may be opened and closed within the same tag, by including a slash before the end of the tag like this: `
`. The introduction of this short-hand, undefined in any HTML 4.01 DTD, may confuse earlier software unfamiliar with this new convention.

To understand the subtle differences between HTML and XHTML consider the transformation of a valid and well-formed XHTML 1.0 document into a valid and well-formed HTML 4.01 document. To make this translation requires the following steps:

1. **The language for an element should be specified with a `lang` attribute rather than the XHTML `xml:lang` attribute** XHTML uses XML's built in language defining functionality attribute.
2. **Remove the XML namespace (`xmlns=URI`)** HTML has no facilities for namespaces.
3. **Change the document type declaration** from XHTML 1.0 to HTML 4.01. (see DTD section for further explanation).
4. If present, **remove the XML declaration** (Typically this is: `<?xml version="1.0" encoding="utf-8" ?>`).
5. **Ensure the document's mime type is set to text/html** In an XHTML document, this may come from the HTTP header sent by the server or from the XML declaration at the start of the document. In an HTML document, this may come from the HTTP header sent by the server or a `<meta>` element within the HTML.
6. **Change the XML empty element short-cut to an HTML style empty element** (`
` to `
`)

Those are the main changes necessary to translate a document from XHTML 1.0 to HTML 4.01. To translate from HTML to XHTML would also require the addition of any omitted opening or closing tags. Whether coding in HTML or XHTML it may just be best to always include the optional labels within an HTML document rather than remembering which labels can be omitted.

A well-formed XHTML document adheres to all the syntax requirements of XML. A valid document adheres to the content specification for XHTML, which describes the document structure.

The W3C recommends several conventions to ensure an easy migration between HTML and XHTML (see HTML Compatibility Guidelines (<http://www.w3.org/TR/xhtml1/#guidelines>)). The following steps can be applied to XHTML 1.0 documents only:

- Including both `xml:lang` and `lang` attributes on any elements assigning language.
- Using the self-closing tag only for elements specified as empty in HTML
- Including an extra space in self-closing labels: for example `
` instead of `
`
- Including explicit close labels for elements that permit content but are left empty (for example, "``", not "``")
- Omit the XML declaration

Note that by carefully following the W3C's compatibility guidelines, a user agent should be able to interpret the document equally as HTML or XHTML. For documents which are XHTML 1.0 and have been made compatible in this way, the W3C permits them to be served either as HTML (with a `text/html` MIME type), or as XHTML (with an `application/xhtml+xml` or `application/xml` MIME type). When delivered as XHTML, browsers should use an XML parser, which adheres strictly to the XML specifications for parsing the document's contents.

Transitional versus Strict

The latest SGML-based specification HTML 4.01 and the earliest XHTML version include three sub-specifications: Strict, Transitional (once called Loose), and Frameset. The Strict variant represents the

standard proper, whereas the Transitional and Frameset variants were developed to assist in the transition from earlier versions of HTML (including HTML 3.2). The Transitional and Frameset variants allow for presentational markup whereas the Strict variant encourages the use of style sheets through its omission of most presentational markup.

The primary differences which make the Transitional variant more permissive than the Strict variant (the differences are the same in HTML 4 and XHTML 1.0) are:

- **A looser content model**
 - Inline elements and plain text (#PCDATA) are allowed directly in: `body`, `blockquote`, `form`, `noscript` and `noframes`
- **Presentation related elements**
 - `underline` (`u`)
 - `strike-through` (`s` and `strike`)
 - `center`
 - `font`
 - `basefont`
- **Presentation related attributes**
 - `background` and `bgcolor` attributes for `body` element.
 - `align` attribute on `div`, `form`, `paragraph` (`p`), and `heading` (`h1`...`h6`) elements
 - `align`, `noshade`, `size`, and `width` attributes on `hr` element
 - `align`, `border`, `vspace`, and `hspace` attributes on `img` and `object` elements
 - `align` attribute on `legend` and `caption` elements
 - `align` and `bgcolor` on `table` element
 - `nowrap`, `bgcolor`, `width`, `height` on `td` and `th` elements
 - `bgcolor` attribute on `tr` element
 - `clear` attribute on `br` element
 - `compact` attribute on `dl`, `dir` and `menu` elements
 - `type`, `compact`, and `start` attributes on `ol` and `ul` elements
 - `type` and `value` attributes on `li` element
 - `width` attribute on `pre` element
- **Additional elements in Transitional specification**
 - `menu` list (no substitute, though unordered list is recommended; may return in XHTML 2.0 specification)
 - `dir` list (no substitute, though unordered list is recommended)
 - `isindex` (element requires server-side support and is typically added to documents server-side)
 - `applet` (deprecated in favor of `object` element)
- The `pre` element does not allow: `applet`, `font`, and `basefont` (elements not defined in strict DTD)
- **The `language` attribute on `script` element** (presumably redundant with `type` attribute, though this is maintained for legacy reasons).
- **Frame related entities**
 - `frameset` element (used in place of `body` for frameset DTD)
 - `frame` element
 - `iframe`
 - `noframes`
 - `target` attribute on `anchor`, client-side `image-map` (`imagemap`), `link`, `form`, and `base` elements

Frameset versus transitional

In addition to the above transitional differences, the frameset specifications (whether XHTML 1.0 or HTML 4.01) specifies a different content model:

```
<html>
  <head>
    Any of the various head related elements.
  </head>

  <frameset>
    <frame></frame>

    <noframes></noframes>
  </frameset>
</html>
```

Summary of flavors

As this list demonstrates, the loose flavors of the specification are maintained for legacy support. However, contrary to popular misconceptions, the move to XHTML does not imply a removal of this legacy support. Rather the X in XML stands for extensible and the W3C is modularizing the entire specification and opening it up to independent extensions. The primary achievement in the move from XHTML 1.0 to XHTML 1.1 is the modularization of the entire specification. The strict version of HTML is deployed in XHTML 1.1 through a set of modular extensions to the base XHTML 1.1 specification. Likewise someone looking for the loose (transitional) or frameset specifications will find similar extended XHTML 1.1 support (much of it is contained in the legacy or frame modules). The modularization also allows for separate features to develop on their own timetable. So for example XHTML 1.1 will allow quicker migration to emerging XML standards such as MathML (a presentational and semantic math language based on XML) and XForms — a new highly advanced web-form technology to replace the existing HTML forms.

In summary, the HTML 4.01 specification primarily reined in all the various HTML implementations into a single clear written specification based on SGML. XHTML 1.0, ported this specification, as is, to the new XML defined specification. Next, XHTML 1.1 takes advantage of the extensible nature of XML and modularizes the whole specification. XHTML 2.0 will be the first step in adding new features to the specification in a standards-body-based approach.

Hypertext features not in HTML

HTML lacks some of the features found in earlier hypertext systems, such as typed links, transclusion, source tracking, fat links, and more.^[18] Even some hypertext features that were in early versions of HTML have been ignored by most popular web browsers until recently, such as the link element and in-browser Web page editing.

Sometimes Web services or browser manufacturers remedy these shortcomings. For instance, members of the modern social software landscape such as wikis and content management systems allow surfers to edit the Web pages they visit.

References

1. ^ HTML Tags (<http://www.w3.org/History/19921103-hypertext/hypertext/WWW/MarkUp/Tags.html>). World Wide Web Consortium. Retrieved on 2007-04-08.
2. ^ First mention of HTML Tags on the www-talk mailing list (<http://lists.w3.org/Archives/Public/www-talk/1991SepOct/0003.html>). World Wide Web Consortium (1991-10-29). Retrieved on 2007-04-08.
3. ^ Index of elements in HTML 4 (<http://www.w3.org/TR/1999/REC-html401-19991224/index/elements>). World Wide Web Consortium (1999-12-24). Retrieved on 2007-04-08.
4. ^ Tim Berners-Lee (1991-12-09). Re: SGML/HTML docs, X Browser (archived www-talk mailing list post) (<http://lists.w3.org/Archives/Public/www-talk/1991NovDec/0020.html>). Retrieved on 2007-06-16. "SGML is very general. HTML is a specific application of the SGML basic syntax applied to hypertext documents with simple structure."
5. ^ ^{a b} Raymond, Eric. "IETF and the RFC Standards Process (http://www.faqs.org/docs/artu/ietf_process.html)", *The Art of Unix Programming* (<http://www.faqs.org/docs/artu>). "In IETF tradition, standards have to arise from experience with a working prototype implementation — but once they become standards, code that does not conform to them is considered broken and mercilessly scrapped. ...Internet-Drafts are not specifications, and software implementers and vendors are specifically barred from claiming compliance with them as if they were specifications. Internet-Drafts are focal points for discussion, usually in a working group... Once an Internet-Draft has been published with an RFC number, it is a specification to which implementers may claim conformance. It is expected that the authors of the RFC and the community at large will begin correcting the specification with field experience."
6. ^ HTML+ Internet-Draft - Abstract (https://datatracker.ietf.org/public/idindex.cgi?command=id_detail&id=789). "Browser writers are experimenting with extensions to HTML and it is now appropriate to draw these ideas together into a revised document format. The new format is designed to allow a gradual roll over from HTML, adding features like tables, captioned figures and fill-out forms for querying remote databases or mailing questionnaires."
7. ^ RFC 1866: Hypertext Markup Language - 2.0 - Acknowledgments (<http://www.ietf.org/rfc/rfc1866.txt>). Internet Engineering Task Force (2005-09-22). Retrieved on 2007-06-16. "Since 1993, a wide variety of Internet participants have contributed to the evolution of HTML, which has included the addition of in-line images introduced by the NCSA Mosaic software for WWW. Dave Raggett played an important role in deriving the forms material from the HTML+ specification. Dan Connolly and Karen Olson Muldrow rewrote the HTML Specification in 1994. The document was then edited by the HTML working group as a whole, with updates being made by Eric Schieler, Mike Knezovich, and Eric W. Sink at Spyglass, Inc. Finally, Roy Fielding restructured the entire draft into its current form."
8. ^ RFC 1866: Hypertext Markup Language - 2.0 - Introduction (<http://www.ietf.org/rfc/rfc1866.txt>). Internet Engineering Task Force (2005-09-22). Retrieved on 2007-06-16. "This document thus defines a HTML 2.0 (to distinguish it from the previous informal specifications). Future (generally upwardly compatible) versions of HTML with new features will be released with higher version numbers."
9. ^
10. ^ HyperText Markup Language Specification Version 3.0 (<http://www.w3.org/MarkUp/html3/CoverPage>). Retrieved on 2007-06-16.
11. ^ Extensions to HTML 3.0 (http://wp.netscape.com/assist/net_sites/html_extensions_3.html). Netscape. "Netscape remains committed to supporting HTML 3.0. To that end, we've gone ahead and implemented several of the more stable proposals, in expectation that they will be approved. ...In addition, we've also added several new areas of HTML functionality to Netscape Navigator that are not currently in the HTML 3.0 specification. We think they belong there, and as part of the standards process, we are proposing them for inclusion."
12. ^ ^{a b} Arnoud Engelfriet. Introduction to Wilbur (<http://htmlhelp.com/reference/wilbur/intro.html>). Web Design Group. Retrieved on 2007-06-16.
13. ^ IETF HTML WG (<http://www.w3.org/MarkUp/HTML-WG/>). Retrieved on 2007-06-16. "NOTE: This working group is closed"
14. ^ <http://www.w3.org/TR/html401/intro/sgmltut.html#h-3.2.2>
15. ^ <http://www.w3.org/TR/xhtml1/diffs.html#h-4.4>

16. ^ <http://www.cs.tut.fi/~jkorpela/qattr.html>
17. ^ See e.g., [XHTML#Relationship_to_HTML](#)
18. ^ Jakob Nielsen (2005-01-03). [Reviving Advanced Hypertext](http://www.useit.com/alertbox/20050103.html) (<http://www.useit.com/alertbox/20050103.html>). Retrieved on 2007-06-16.

See also

- [Web Hypertext Application Technology Working Group](#)
- [Alt attribute](#)
- [Character encodings in HTML](#)
- [List of XML and HTML character entity references](#)
- [Cascading Style Sheets](#)
- [Dynamic HTML](#)
- [*The HTML Sourcebook: The Complete Guide to HTML* \(historical reference from 1995\)](#)
- [HTML editor](#)
- [HTML element](#)
- [HTML scripting](#)
- [Microformats](#)
- [Parsing](#)
- [Tim Berners-Lee](#)
- [Unicode and HTML](#)
- [Web colors](#)
- [List of document markup languages](#)
- [HTML5](#)
- [Comparison of document markup languages](#)
- [Comparison of layout engines \(HTML\)](#)
- [Comparison of layout engines \(HTML5\)](#)
- [XHTML](#)
- [Wikipedia:WikiProject Usability/HTML](#)

External links

Tutorials and guides

- [HTML Source: HTML Tutorials \(<http://www.yourhtmlsource.com/>\)](#)
- [HTML.net \(<http://www.html.net/tutorials/html/introduction.asp>\)](#)
- [HTMLQuick.com tutorials \(<http://www.htmlquick.com/tutorials.html>\)](#)

HTML Markup Validators

- [W3C's Markup Validator \(<http://validator.w3.org/>\)](#)
- [WDG HTML Validator \(<http://www.htmlhelp.com/tools/validator/>\)](#)
- [Off-line HTML Validator v1.0 \(<http://kzone.host.sk/download/item4.htm>\) for Windows](#)

Standard HTML specifications

- Refer to the [HTML version history](#) section

Other specifications

- Web Applications 1.0 (<http://whatwg.org/specs/web-apps/current-work/>) A specification generally referred to as "HTML 5". The Web Hypertext Application Technology working group are an independent initiative who cooperate with the W3C.

Retrieved from "<http://en.wikipedia.org/wiki/HTML>"

Categories: Wikipedia proseline cleanup | HTML | Computing acronyms | ISO standards | Markup languages | Technical communication | W3C standards

- This page was last modified 20:29, 27 August 2007.
- All text is available under the terms of the GNU Free Documentation License. (See [Copyrights](#) for details.) Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a US-registered 501(c)(3) tax-deductible nonprofit charity.